

WHAT IS CLAIMED IS:

1. A method for preventing deadlock in communication between a user and a network interface card (NIC), the method comprising steps of:
 - a. writing a doorbell associated with at least one descriptor having a descriptor QP context to a buffer in the NIC,
 - b. if said buffer is full, dropping at least one doorbell from said buffer, thereby allowing a write of a new doorbell to said buffer, and
 - c. if said buffer is not full, executing each descriptor associated with said written doorbell in order of descriptor posting by the user,
whereby the method enables unrestricted user-level access to the NIC.
2. The method of claim 1, wherein said step of executing each descriptor is preceded by the step of reading said QP context to verify access rights for said descriptor, thereby providing a QP context read response.
3. The method of claim 1, wherein said step of dropping at least one doorbell from said buffer is followed by steps of:
 - d. recovering each said dropped doorbell, thereby obtaining a recovered doorbell; and
 - e. executing each descriptor associated with said recovered doorbell.
4. The method of claim 1, wherein said step of executing each descriptor further includes:
 - i. storing a respective doorbell associated with each said descriptor in said doorbell buffer,
 - ii. reading said QP context of each said descriptor,
 - iii. checking if said doorbell is a repeat doorbell, and if no,
 - iv. executing each said descriptor.

5. The method of claim 3, wherein said step of writing a doorbell is preceded by steps of writing said at least one descriptor to memory and updating a doorbell record, and wherein said step of recovering includes:

- i. checking if a doorbell was dropped, and if yes,
- ii. reading said doorbell record,
- iii. checking if a software doorbell counter is larger than a hardware doorbell counter, and if yes,
- iv. scheduling the execution of one or more descriptors on a relevant queue pair associated with said doorbell record.

6. The method of claim 5, wherein said buffer is a first-in first-out (FIFO) buffer.

7. The method of claim 4, wherein said step of executing each descriptor further includes dropping repeat doorbells found in said checking if said doorbell is a repeat doorbell.

8. The method of claim 1, wherein said NIC is connected to an InfiniBand fabric, and wherein said at least one descriptor is a work queue element (WQE).

9. A method for resolving communication deadlock arising from overlapping doorbell write and context read response paths, comprising steps of:

- a. providing a single logical communication path common to doorbell writes and context read responses, and
- b. facilitating unsynchronized, kernel-call-free, unrestricted traffic along said single common path,

whereby the method enables unrestricted user-level access to a network interface adapter without having to use kernel calls.

10. The method of claim 8, wherein said step of facilitating unsynchronized, kernel-call-free, unrestricted traffic along said single common path includes:

- i. writing a doorbell associated with at least one descriptor to a doorbell buffer in said NIC,
 - ii. dropping at least one doorbell from said doorbell buffer, thereby providing space in said buffer for at least one new doorbell, and
 - iii. recovering each dropped doorbell and executing its respective associated at least one descriptor.
11. The method of claim 10, wherein said step of dropping at least one doorbell occurs in response to a first check that indicates said doorbell buffer is full.
12. The method of claim 10, wherein said step of recovering each dropped doorbell includes recovering each dropped doorbell from a system memory.
13. The method of claim 10, wherein said doorbell buffer is a first-in first-out (FIFO) buffer, and wherein said dropping of at least one doorbell from said doorbell buffer includes dropping a last doorbell input into said FIFO buffer.
14. The method of claim 10, wherein said buffer is a first-in first-out (FIFO) buffer, wherein said writing a doorbell associated with at least one descriptor to a doorbell buffer includes writing a last doorbell to said FIFO buffer, and wherein said dropping of at least one doorbell from said doorbell buffer includes dropping a first doorbell from said FIFO buffer.
15. A system that enables unrestricted user-level access to a network interface card (NIC), comprising:
- a. a NIC configured to accept doorbell rings and context read responses through a single logical path,
 - b. at least one host central processing unit (CPU) running at least one application, said at least one CPU connected to the NIC through an interface bus, said application operative to write descriptors associated with said doorbell rings and to

update a doorbell record in a system memory, said single logical path passing through said interface bus; and

c. a kernel call-free mechanism for facilitating free traffic along a said single logical path,

whereby said NIC configuration removes the need for a separate context storage memory attached to the NIC and whereby said kernel call-free mechanism allows a practically unlimited number of users to access the NIC simultaneously without a kernel call.

16. The system of claim 15, wherein said NIC configuration includes a buffer for temporarily storing said doorbells, and wherein said kernel call-free mechanism includes a doorbell dropping mechanism for dropping at least one doorbell from said buffer if said buffer is full, and a recovery mechanism for recovering dropped doorbells and for executing their respective associated descriptors.

17. The system of claim 16, wherein said buffer is a first-in first-out (FIFO) buffer.

18. The system of claim 16, wherein said doorbell recovery mechanism includes a software doorbell counter and a hardware doorbell counter, and means to compare between said two counters.

19. A method for obtaining kernel call-free access from a plurality of users that post doorbells to a doorbell buffer in a network interface adapter, the adapter having to obtain read responses from a system memory to allow execution of descriptors associated with each such doorbell, the method comprising steps of:

a. responsive to a first check, dropping at least one doorbell from the doorbell buffer if the buffer is full, thereby providing space in the buffer for a respective at least one new doorbell;

b. recovering each dropped doorbell and executing its respective associated descriptors; and

c. responsive to same said first check, if said doorbell buffer is not full, checking if a doorbell is a repeat doorbell, and executing descriptors of each doorbell found to be not a repeat doorbell.